EMOTIC: EEG Monitoring of Thoughts and Individual Conditions

Maisha Supritee Chowdhury[mschowdh], Edan Stasiuk[ews3], Rashed Hadi[rmh7], Zihao Xie[zxa43], and Daniel Kim[tjk5]

{mschowdh,ews3,rmh7,zxa43,tjk5}@sfu.ca
CMPT340 Spring 2024, Prof. Hamarneh

Abstract. This project explores the use of machine learning (ML) techniques to classify human emotions based on electroencephalogram (EEG) data. Initially, the project aimed to utilize the NeuroSky MindWave Mobile 2 headset for data collection, leveraging its affordability and accessibility. However, due to technical challenges with Bluetooth connection, we pivoted to using publicly available online datasets. The primary goal of this work is to process EEG data, extract meaningful features, and train an ML model capable of predicting the emotional state associated with a given EEG sample. The report details the methodology employed, including data processing, feature engineering, and model selection. Preliminary results demonstrate the potential of ML models in accurately classifying emotions from EEG signals, underscoring the applicability of this approach in affective computing and human-computer interaction.

Keywords: EEG · Emotion Classification · Machine Learning · Feature Extraction · Affective Computing · NeuroSky MindWave · Signal Processing · Human-Computer Interaction · Emotion Recognition · Data Preprocessing

1 Introduction

1.1 Application and Problem Description

The increasing integration of emotion recognition in technologies like affective computing and human-computer interaction (HCI) underscores the importance of accurate emotion classification systems. This project, EMOTIC: EEG Monitoring of Thoughts and Individual Conditions, aims to harness electroencephalogram (EEG) data to classify human emotions through machine learning (ML) techniques. Initially, the project planned to use the NeuroSky MindWave Mobile 2 headset for data collection due to its affordability and accessibility. However, technical challenges in establishing a reliable Bluetooth connection required a pivot to publicly available datasets. The project involves EEG data processing, extracting relevant features, and developing an ML model capable of predicting emotional states from EEG signals. The resulting full-stack web application provides users with an interface to upload EEG data and receive emotional state predictions in real time.

1.2 Motivation

Emotion recognition systems hold transformative potential across various industries including mental health diagnostics, personalized learning environments, and adaptive gaming systems. EEG-based approaches offer a noninvasive and objective means of understanding human emotions, providing deeper insights into mental states compared to traditional methods. However, challenges such as data accessibility, feature extraction, and reliable classification methods remain. This project aims to contribute to this growing field by addressing these challenges and exploring how ML can enhance EEG-based emotion recognition.

1.3 Background

Electroencephalography (EEG) is a widely used technique to monitor brain activity by recording electrical signals along the scalp. Advances in ML have paved the way for processing these complex signals to classify emotional states. Devices like the NeuroSky MindWave Mobile 2 headset have popularized EEG technology due to their affordability and versatility. However, real-world applications often encounter challenges such as device limitations and noisy data. Using online datasets and ML techniques, this project explores an alternative approach, building on existing advancements in affective computing and EEG-based studies.

1.4 Related Work

The YouTube video "EEG Emotion Prediction: Training a Model" [22] served as a foundational resource for this project. This work provided valuable information on the methodology for processing EEG data and training ML models for emotion classification. Our project builds on these ideas, extending the scope to include feature engineering, advanced ML models, and a web-based application for user interaction.

1.5 Map of the Report

The rest of this report is organized as follows. Section 2 describes the data sets used to train the model, including their sources and data processing steps. Section 3 details the experimental procedure, focusing on feature engineering and model training. Section 4 presents the results, including the classification report and the confusion matrices derived from the model. Section 5 outlines our achievements and the challenges faced, particularly the failure to utilize the NeuroSky MindWave Mobile 2 headset. Section 6 describes the contributions of each team member and their role in the project. Section 7 provides a summary of the project, followed by a critical analysis of its outcomes. Finally, Section 8 discusses potential directions for future work, such as manual data collection, incorporating multimodal data, and real-time emotion detection.

2 Materials

For the creation of our emotion detection model, we utilized two datasets for development and training. Below, we detail the datasets and the preprocessing steps undertaken to adapt the data for our requirements.

2.1 EGG Brainwave Dataset: Feeling Emotion

The initial phase of the project utilized a small EEG dataset from Kaggle, comprising of data from one male and one female subject. This dataset served as a valuable starting point for setting up the pipeline and training the model. However, its limited size made it unsuitable for meaningful training or robust model evaluation. Consequently, we conducted extensive research to identify a dataset better suited to the project's requirements, ultimately selecting the FACED dataset.

2.2 Finer-grained Affective Computing EEG Dataset (FACED)

The FACED dataset is a comprehensive collection of EEG recordings of 123 subjects designed for emotion detection. Each EEG data file corresponds to a single subject and contains an array with dimensions (28, 32, 7500), where:

- 28: The number of videos shown to subjects.
- 32: The number of EEG channels recorded.
- 7500: The number of time points, representing 30 seconds of data recorded at a sampling frequency of 250 Hz.

The videos are categorized into nine distinct emotional states:

- Neutral: Neutral emotion.
- Positive Emotions: Amusement, inspiration, joy, tenderness.
- Negative Emotions: Anger, disgust, fear, sadness.

Although the FACED dataset was ideal for our objectives, its size posed challenges for efficient training and analysis. To address this, we reduced the number of time points per subject by a factor of 1/50, creating arrays of dimensions (28, 32, 150). This reduction allowed us to work with the dataset while preserving sufficient temporal information for emotion classification. This preprocessing was conducted using the Pickle Python library.

3 Methods

3.1 Experiment procedure

In order to collect data for the experiment, the group had originally devised a plan to gather willing participants and place them in a controlled environment, before equipping subjects with an EEG headset. However, due to issues with the MindWave Mobile 2 headset, (**Section 5**), our approach pivoted to using an open-access EEG dataset (FACED).

We started by setting up a Python virtual environment to install all the necessary packages for the model. This approach helped isolate the libraries and dependencies required for the project, preventing conflicts with pre-existing packages on local machines.

The following packages were required for the project: NumPy, pandas, TensorFlow, Matplotlib, scikit-learn, seaborn, Flask, and flask_cors.

3.2 Data Preprocessing

The dataset, provided as pickle files (.pkl), was preprocessed as follows by process.py:

– Converted to a NumPy array and downsized by reducing timepoints by a factor of 50. The original dataset shape of (100, 28, 32, 7500) was transformed into (100, 28, 32, 150).
– Reshaped into (2800, 4800), where the number of channels was multiplied with the reduced timepoints.

3.3 SVM Model

The machine learning algorithm that provided the best success for our use case was the Support Vector Machine. We discovered that it handled the format of EEG data very well, considering that there were 32 channels to be analyzed in parallel. This was because the model was equipped with the ability to handle multi-dimensional features very well.

– Input shape: A 2D NumPy array, formatted as [100 subjects * 28 videos, 32 channels * 150 time points]
– Configurations: Kernel set to 'linear', with C value set to 2.0. This was important for balancing the tradeoffs between maximizing margins and minimizing classifications.
– Techniques: Standardized data before training to improve model performance. Utilized an 80:20 train-to-test ratio. Performed optimization on the random state parameter to find the seed with the highest accuracy.

3.4 GRU Model

This model is using Tensorflow's recurrant neural network, specifically gated recurrent unit. In model.py, the reshaped data from process.py was converted into a pandas DataFrame with labels appended as the last column. The dataset was split into 80% training and 20% testing using scikit-learn, producing: x_train, y_train for training, x_test, y_test for testing and a labelMap was created for reference during evaluation.

Model Design The buildModel function constructed the GRU-based model using the following setup:

- Input shape: The 4800 columns from x_train, reshaped to 3D for GRU.
- Architecture: GRU layers with 64 units for processing temporal features.
- Optimizer: Adam
- Loss Function: Sparse Categorical Crossentropy (ideal for multiclass classification).
- Validation Split: 20
- Epochs: 25 (reduced from 50 due to time constraints).
- Early Stopping: Patience of 10 epochs to halt training and restore the best weights if loss did not improve.
- The trained model was saved as a Keras file.

The saved model was reloaded for testing using x_test. The results function performed the following:

- Evaluated the model on x_test to retrieve loss and accuracy metrics.
- Generated a classification report and confusion matrix for detailed performance insights.
- Compiled all evaluation results into JSON format for reporting.

4 Results

As previously mentioned, we completed our detection model using the Support Vector Machine (SVM) model. The evaluation, conducted on a test set of 20 subjects, demonstrated an overall accuracy of 78.21%, underscoring the model's ability to classify the majority of samples correctly. For detailed analysis, refer to the classification report in Appendix A, which illustrates the model's effectiveness. Additionally, the confusion matrix, found in Appendix B, provides further insight into the model's performance across different classes.

5 Accomplishments

We successfully developed a full-stack web-based application capable of predicting emotions from EEG data using a machine learning model. Our project initially aimed to collect EEG data using the NeuroSky MindWave Mobile 2 headset. Unfortunately, technical issues prevented us from using the device. We ultimately succeeded with the FACED datasets.

5.1 Connection Attempts on macOS

We attempted to connect the headset to a MacBook running macOS Monterey 12.6.3 by following the official guide for versions 10.8–10.13:

1. Power on the headset and ensure the blue LED indicator was active.

2. Pair via *System Preferences* → *Bluetooth*, select *MindWave Mobile*, and click *Pair*.
3. Download the setup kit from `MWM2.neurosky.com`.
4. Launch the tutorial app and follow the instructions.

Though pairing was successful, the app failed to recognize the headset. After ensuring the headset was powered and trying multiple connections, the issue persisted.

We found an archived instructional video, but it only repeated the existing steps. We also explored the `python-mindwave` repository, which relied on the ThinkGear Connector (TGC) for communication. Despite reinstalling the TGC and creating the required port `/dev/tty.MindWaveMobile`, we encountered the persistent error: *"Headset signal noisy 255. Adjust the headset and the earclip."* Despite attempts to resolve the issue, a stable connection couldn't be established.

Additionally, we searched for the "MindSet Development Tools," but these were unavailable, complicating troubleshooting efforts.

5.2 Connection Attempts on Windows

We tested the headset on a laptop running Windows 11, borrowed from the Simon Fraser University library. The quick start guide provided instructions for Windows 10, leading to compatibility issues. Following the steps:

1. Turn on the headset and confirm the blue LED.
2. Pair via *Settings* → *Devices* → *Bluetooth*.
3. Enter the passkey `0000`.
4. Download and run the tutorial app from `MWM2.neurosky.com`.

While pairing was successful, the connection was immediately lost, and administrative privileges were required to install the tutorial software, which were unavailable.

5.3 Connection Attempts on iOS

Attempts to connect on iOS devices were unsuccessful. The MindWave Mobile app either crashed due to being outdated or failed to recognize the headset on older devices.

5.4 Challenges and Conclusions

Despite troubleshooting, we could not establish a functional connection with the headset. Contributing factors likely included outdated software, OS compatibility issues, and missing instructional resources. Moreover, security concerns arose when bypassing system protections to install necessary software.

We also faced difficulties generating appropriate sample CSV files for the GRU model, ultimately leading to a switch to the SVM model. These challenges emphasized the difficulties of working with outdated hardware and software in a modern development environment, prompting us to shift to alternative datasets and models for progress.

6 Contributions

Maisha Supritee Chowdhury:

- Researched different machine learning models for EEG analysis, tested different model training methods to come up with appropriate model for our case.
- Responsible for implementing the Gated-Recurrent-Unit model to perform classifications on sample EEG data, with the help of Gabriel Atkin's EEG Emotion Prediction video.
- Implemented code for processing eeg dataset in collaboration with Daniel Kim.
- Improved GRU model, further processed dataset for GRU, and created classification report and confusion matrixes.
- Investigated different ways to batch process dataset as the GRU model was taking too long, wrote code to predict emotions with the GRU model.
- Collaborated with Daniel Kim and Edan Stasiuk to clean-up and document the project's directory structure.
- Worked with Daniel Kim to tweak his SVM model to classify, and bridge the model with Edan Stasiuk's and Zihao Xie's backend Flask code.
- Designed frontend using Figma, in collaboration with Edan Stasiuk.
- Wrote weekly meeting notes and worked on README instructions for reproduction of project with Daniel Kim and Edan Stasiuk.

Edan Stasiuk:

- Conducted research on SDK availability and compatibility for the MindWave Mobile 2, locating relevant documentation for .NET development.
- Discovered additional resources, including an Xcode SDK, for the device, and JetBrains' Rider IDE as a viable solution for .NET development on macOS.
- Investigated alternative, more reliable devices to the MindWave Mobile 2, such as the Emotiv Insight Headset, BrainBit Headband, and Muse 2.
- Designed the frontend using Figma, collaborating with Maisha Supritee Chowdhury on the creation of graphs and information displays for the landing and app pages.
- Developed the frontend using Next.js, TypeScript, and Tailwind CSS to implement the design.
- Collaborated with Rashed to establish a Bluetooth connection with the MindWave Mobile 2, utilizing a Windows machine he provided, which was rented from the SFU library.
- Integrated the Flask backend code, building upon the foundation created by Zihao Xie, who developed an endpoint for uploading and saving data from the frontend. An additional endpoint was created to send processed data back to the frontend, ensuring seamless functionality of the full-stack application. The backend code for data processing was provided by Daniel Kim and Maisha Chowdhury.

- Collaborated with Maisha Chowdhury and Daniel Kim to clean-up and document the project's directory structure.

Rashed Hadi:

- Explored ways to use the MindWave Mobile 2 to gather EEG data, collaborated with Edan Stasiuk to attempt establishing a connection with the headset on various platforms.
- Attempted to solve MindWave Mobile 2 connectivity issues using NeuroSky social media platforms, whilst attempting to contact NeuroSky support.
- Gathered and organized subjects for EEG data collection on multiple occasions that had to be canceled due to MindWave Mobile 2 connectivity issues.
- Researched and filtered various potential publicly available EEG datasets for model training alongside Daniel Kim.
- Processed data from the FACED dataset to get it ready for model training.
- Created a script to plot EEG data from the FACED dataset for reporting.
- Researched and identified ways the ML model can be hosted.
- Created the demo video.

Zihao Xie:

- Conducted in-depth research on machine learning models for EEG analysis, assessing their suitability using publicly available datasets.
- Curated and pre-processed EEG datasets to ensure compatibility with machine learning models.
- Utilized Flask for backend development, enabling EEG CSV file uploads and data preparation for machine learning processing, following suggestions from Edan Stasiuk.
- Developed and optimized Flask endpoints to manage data processing requests efficiently, ensuring seamless integration with the machine learning models.
- Collaborated with Edan Stasiuk to design the communication flow between the backend and frontend, implementing the backend-to-frontend communication to ensure accurate transmission of processed EEG results.

Daniel Kim:

- Responsible for implementing the Support-Vector-Machine program to perform classifications on sample EEG data.
- Experimented with different machine learning methodologies, including the K-Nearest-Neighbors algorithm. Studied the outcomes to make improvements, eventually creating the SVM model.
- Performed Python scripting to create data pre-processing and down-sampling functions, along with a script to generate random samples for each emotion.
- Researched the methodologies involved in sampling, analyzing, and training EEG data on machine learning models.
- Researched and compared multiple sources of open-access data, resulting in the discovery of the FACED dataset.
- Worked alongside Rashed Hadi to parse data from the FACED dataset, allowing fluid integration into Maisha Chowdhury's GRU model.

7 Conclusion and Discussions

In conclusion, this project provided us with the opportunity to develop an interactive platform that enables users to utilize our emotion detection model for analyzing EEG data. While the development of a functional detection model presented challenges, these obstacles were integral to our learning experience and contributed to the uniqueness of the project.

Throughout the course of the project, we were required to adapt and adjust our approach when faced with challenges, such as the inability to establish a connection with the MindWave Mobile 2 for data collection. This setback prompted us to reassess our strategy and conduct an extensive search for alternative datasets that met our requirements. Although this pivot introduced significant time constraints as the project deadline approached, it also served as a valuable test of our teamwork, adaptability, and time management skills.

8 Future Work

There are several opportunities for future work that can enhance the accuracy and applicability of our model. The following points highlight the extensions that we believe would benefit this project the most:

8.1 Manual Data Collection

Due to technical limitations with our EEG signal reader, as discussed previously, the model was created using externally sourced EEG data. A significant extension would involve obtaining a more reliable EEG signal reader to collect data in controlled experiments, allowing the tests to be tailored to meet specific model training needs. This approach would ensure the model is trained on data closely aligned with its intended application, leading to improved accuracy and robustness.

8.2 Incorporating Multimodal Data

Combining EEG data with other physiological signals, such as heart rate variability, skin conductance, or facial expressions, would be another significant extension. This multimodal approach would allow the model to cross-validate emotional states and compensate for noise or inaccuracies that lie in each data stream.

8.3 Real-Time Emotion Detection

An experimental extension that would take the model's applicability to another level is implementing real-time emotion detection. This would involve optimizing the model to process EEG signals with minimal latency, as well as exploring hardware and software integrations that would allow for seamless operation in the real world. Applications would include mental health monitoring, gaming, and virtual reality.

These extensions, along with the many other ways this model can be extended, would allow for the creation of a robust and highly applicable emotion detection model that can be used across many fields.

9 Acknowledgments

We would like to express our gratitude to the creators of the YouTube video "EEG Emotion Prediction: Training a Model" [22], which provided valuable guidance on training machine learning models using EEG data. The insights and methodologies presented in the video greatly contributed to the development and training of our emotion prediction model. We appreciate the effort and expertise shared by the video's creators, which significantly enhanced the effectiveness of our project.

Appendix

A

SVM Model Classification Report:

Label	Precision	Recall	F1 Score	Support
Amusement	0.79	0.89	0.84	61
Anger	0.87	0.83	0.85	65
Disgust	0.68	0.75	0.71	63
Fear	0.77	0.74	0.76	69
Inspiration	0.77	0.83	0.8	58
Joy	0.85	0.83	0.84	64
Neutral	0.72	0.71	0.71	72
Sadness	0.79	0.69	0.74	59
Tenderness	0.81	0.8	0.8	49
Accuracy			0.78	560
Macro Average	0.79	0.78	0.78	560
Weighted Average	0.78	0.78	0.78	560

Test Accuracy: 78.21%

Fig. 1. Classification report for the SVM model including a test accuracy value.

B

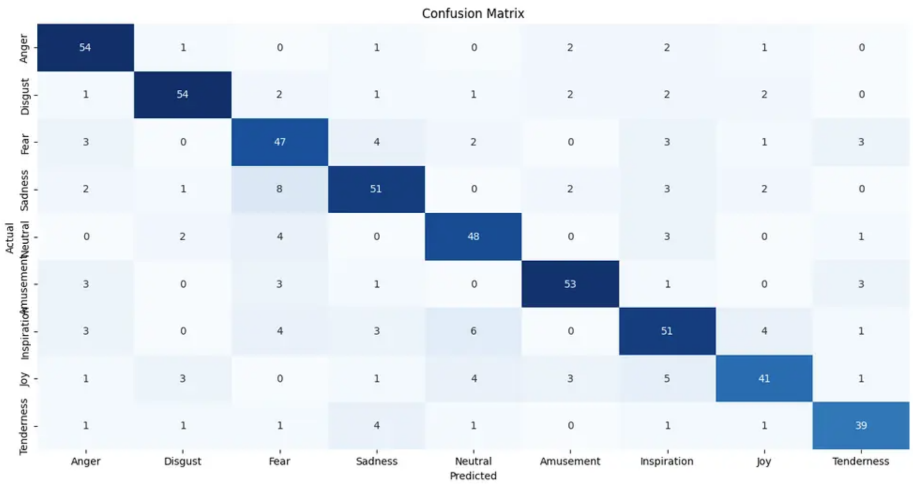

Fig. 2. Confusion matrix for the SVM model.

References

1. S. Aksoy. Getting eeg data from mindwave mobile headset. *Accessed: Oct 23, 2024*, February 2024.
2. E. V. D. Angelakis and P. Asvestas. Brain signals to actions using machine learning. *Engineering Proceedings*, 50(1):7, 2023. Accessed: Oct 29, 2024.
3. G. Baker. Machine learnign: Classification. *CMPT 353*, September 2024. Accessed: Oct 30, 2024.
4. G. Baker. Machine learning: Other techniques. *CMPT 353*, September 2024. Accessed: Oct 30, 2024.
5. M.-K. K. et al. A review on the computational methods for emotional state estimation from the human eeg. *Computational and Mathematical Methods in Medicine*, March 2013. Accessed: Oct 30, 2024.
6. X.-W. W. el al. Emotional state classification from eeg data using machine learning approach. *Neurocomputing*, 129:94–106, 2014. Accessed: Oct 30, 2024.
7. X. C. et al. Emotion recognition of single-electrode eeg based on multi-feature combination in timefrequency domain. *Journal of Physics: Conference Series*, 2021. Accessed: Oct 30, 2024.
8. L. Jorge. Next.js with python backend: Combining the best of both worlds for web development. *Medium*, July 2023. Accessed: Nov 13, 2024.
9. G.-Y. C. et al. Eeg dataset for the recognition of different emotions induced in voice-user interaction. *Scientific Data*, October 2024. Accessed: Nov 17, 2024.
10. J. C. et al. A large finer-grained affective computing eeg dataset. *Scientific Data*, October 2023. Accessed: Nov 20, 2024.

11. T.-T. Tseng. Thinkgear sdk for .net: Development guide and api reference. *ThinkGear SDK Dev Guide*, November 2024. Accessed: Nov 24, 2024.
12. NeuroSky. *NeuroSky MindWave Instruction Manual*. NeuroSky, Inc BV, 2009. Accessed: Nov 15, 2024.
13. Meagmohit. Eeg-datasets. `https://github.com/meagmohit/EEG-Datasets`, 2024. Accessed: Oct 19, 2024.
14. Afiedler. node-thinkgear-sockets. `https://www.npmjs.com/package/node-thinkgear-sockets`, 2015. Accessed: Oct 23, 2024.
15. R. Ramele. Python Mindwave. `https://github.com/faturita/python-mindwave`, January 2023. Accessed: Oct 13, 2024.
16. H. T. M. M. Games. Reading brainwaves using mindwave mobile 2 and unity - part 1. `https://www.youtube.com/watch?v=6lcuj0jYiqQ`, August 2019. Accessed: Nov 24, 2024.
17. DevXplaining. Your brains on python part 2: Python mindwave brainwave recorder and how to get started. `https://www.youtube.com/watch?v=jXMd5aNIIoQ`, 2021. Accessed: Nov 24, 2024.
18. Emotiv. Insight - 5 channel wireless eeg headset. `https://www.emotiv.com/products/insight`, 2024. Accessed: Nov 29, 2024.
19. Brainbit. Brainbit headband. `https://store.brainbit.com/products/brainbit-sdk`, 2024. Accessed: Nov 29, 2024.
20. Muse. Muse 2. `https://ca.choosemuse.com/products/muse-2`, 2024. Accessed: Nov 29, 2024.
21. Dashersw. xwits — a node.js library that reads your mind with emotiv epoc eeg headset. `https://github.com/dashersw/wits`, 2019. Accessed: Oct 29, 2024.
22. G. Atkin. Eeg emotion prediction - data every day. `https://www.youtube.com/watch?v=IAQdqaoHrfE`, December 2020. Accessed: Nov 5, 2024.
23. NeuroSky. Mindwave mobile 2 - brainwave sensing headset. `https://store.neurosky.com/pages/mindwave`, 2015. Accessed: Nov 13, 2024.
24. Miwash. `FearDetectionw/EEGDatawhileplayingVRFnaf`. https://www.kaggle.com/code/miwash/fear-detection-w-eeg-data-while-playing-vr-fnaf/notebook, June 2024. Accessed: Nov 13, 2024.
25. M. R. Khan. Multi class emotion detection. `https://www.kaggle.com/code/khan1803115/multi-class-emotion-detection/notebook#Traing`, March 2024. Accessed: Nov 13, 2024.
26. NeuroSky. Mindwave can't work on mac osx 10.11 or higher. `http://support.neurosky.com/kb/mindwave/mindwave-cant-work-on-mac-osx-1011-or-higher`, January 2017. Accessed: Nov 15, 2024.
27. BCMI. Seed dataset - a dataset collection for various purposes using eeg signals, 2019. Accessed: Nov 17, 2024.
28. J. J. Bird. Eeg brainwave dataset: Feeling emotions. `https://www.kaggle.com/datasets/birdy654/eeg-brainwave-dataset-feeling-emotions`, 2018. Accessed: Nov 17, 2024.
29. M. S. et al. Hci-tagging-database. `https://mahnob-db.eu/hci-tagging/`, 2012. Accessed: Nov 19, 2024.

articles [1,2,3,4,5,6,7,8,9,10,11] book [12] misc [13,14,15,16,17,18,19,20,21,22,23,24,25,26,27,28,29]